第44卷 第4期 四川大学学报(工程科学版) Vol. 44 No. 4
2012年7月 JOURNAL OF SICHUAN UNIVERSITY (ENGINEERING SCIENCE EDITION) July 2012

文章编号: 1009-3087(2012)04-0122-07

投影残差分类器

于传帅[1],冯 勇[1],徐可佳[1],谭治英[1],李玲娜[1,2]
(1. 中国科学院 成都计算机应用研究所,四川 成都 610041; 2. 西南石油大学 理学院,四川 成都 610500)

摘 要: 针对声音、图像等高维数据的分类问题,提出了一种快速算法。首先通过非线性特征映射,将各个类别的训练样本集转换到特征空间中,构造相应的特征子空间,然后提取它们的主要特征。特征映射能够降低特征子空间的维数,并增强它们之间的两两正交性,提高了分类的准确性。在进行分类时,该方法将测试样本向各个特征子空间投影,并计算投影残差,测试样本即为投影残差最小的特征子空间的样本。与传统的分类方法不同,快速算法能一次区分多个类别,并具有与支持向量机相同的准确率。又使用了流形学习理论对快速算法进行改进,在保持准确率的前提下,极大地降低了特征子空间的维数,验证了流形学习理论的应用价值。

关键词: 核方法; 特征子空间; 投影残差; 流形学习
中图分类号: O235 **文献标志码:** A

Projection Residual Classifier

YU Chuan-shuai[1], *FENG Yong*[1], *XU Ke-jia*[1], *TAN Zhi-ying*[1], *LI Ling-na*[1,2]
(1. Chengdu Inst. of Computer Applications, CAS, Chengdu 610041, China; 2. School of Sciences, Southwest Petroleum Univ., Chengdu 610500, China)

Abstract: A fast algorithm for the classification of high dimensional vectors was proposed. A special nonlinear feature function was used to reshape training sample sets of multi classes to low dimensional and orthogonal feature sub-spaces. Then the principle components of each feature sub-space were calculated. By projecting a new coming vector on each feature sub-space the projection residuals was calculated. The new vector was regarded as a sample of the feature sub-space with the smallest residual. This algorithm can distinguish high dimensional vectors of multi classes by one comparison and has good accuracy. Furthermore, manifold learning theory was added to the feature function to keep the accuracy and greatly reduce the dimensionality of feature sub-spaces.

Key words: kernel method; feature sub-space; projection residual; manifold learning

人类在日常生活中会遇到各种各样的声音、图像等高维数据,并对这些数据进行识别、区分。使用计算机模拟这一过程,对数据进行分类也是模式识别领域的一个重要问题,分类算法被简称为分类器。目前广泛使用的分类器有 Bayes[1]、KNN[2]、Adaboost[3]、SVM[4-6]、LDA[7]。其中 Bayes 分类器利用后验概率进行分类,KNN 分类器利用样本点邻近的 k 个训练样本进行分类,Adaboost 分类器能够把几种弱分类器组合成一个强分类器。SVM 分类器、LDA分类器是现阶段比较流行的分类器,SVM分类器致力于在两个类别之间建立分类间隔,处在间隔边界上的训练样本被称为支持向量,而 LDA 分类器则寻找两个类别差异最大的方向,将样本在这个方向上投影之后再进行分类。它们的出发理念都是寻找线性分隔,一次只能区分两个类别,在引入核方法后成为非线性分类器。

核方法能够将很多线性统计分析工具衍生为非线性统计工具,它先使用非线性特征映射 $\varphi: R^N \to F$ 将原空间映射到特征空间,然后再在特征空间中做各种线性统计分析,返回到原空间就是非线性统计分析。如果这些分析只需要特征空间中的内积运算 $\langle \varphi(x_1), \varphi(x_2) \rangle$,Mercy 定理[8] 就能保证这个运算可以用一个半正定的核函数 $k(x_1, x_2) = \langle \varphi(x_1), \varphi(x_2) \rangle$ 来代替,这样通过核函数就能进行非线性统计分析。

收稿日期: 2012-01-12
基金项目: 国家"973"计划资助项目(2011CB302400); 国家自然科学基金资助项目(10771205); 中国科学院西部之光资助项目
作者简介: 于传帅(1983—),男,博士. 研究方向: 图像处理; 核方法; 流形学习. E-mail: yuchuanshuai@yahoo.com.cn

在实际应用中,如果能够有效的提取每个类别的主要特征,就能利用这些特征建立一种高效的分类器。以图像为例,一个物体的所有图像是一个无穷的集合,这些图像受到光照、物体的平移、拍摄的角度、拍摄点离物体的距离等因素影响,虽然每个图像的维度非常高,但是整个集合却只有光照、平移、角度、距离等这几个维度。流形学习是识别高维数据集的内在维度的方法,比较经典的流形学习方法有 MDS[9]、LLE[10]、Isomap[11]、SDE[12-13]、LE[14] 等,流形学习只能将数据集的内在维度可视化,但无法计算每个维度的函数表达式。在 PCA 和核方法基础上演变而来的 KPCA[15] 可以实现这一过程,它在特征空间中做 PCA 来提取特征空间的主成分特征。

流形学习与 KPCA 实际是一个问题的两个方面[16-17],它们都通过非线性变换来提取每个类别的主要特征。流形学习虽然效果好,但没有具体的函数表达式,KPCA 有具体的函数表达式,但效果不是很理想,如果将流形学习和核方法结合起来,就能提高实验效果。目前还有很多优化核函数的文献[18-20],可以提高核方法的效果。

作者在这些理论基础上建立了投影残差分类方器,该算法能够一次区分多个类别,速度快,识别率高,提出了将多个类别问题转化为低维的、两两正交的特征子空间的观点,并使用投影残差来进行分类。然后将流形学习方法添加到分类器,使得特征子空间维度的急剧减少,验证了流形学习理论的应用价值。最后还提出优化核函数与多类问题的流形学习理论框架。

1　投影残差分类器

在 KPCA 理论的基础上,首先建立一种基于投影残差的快速算法。以图像为例,首先使用与 Guassian 核函数相对应的非线性特征映射来改变每个物体的图像集合,使得这些集合成为低维的、两两正交的特征子空间。这样一方面各个特征子空间更容易区分了,另一方面由于维度的降低运算速度也提高了。然后使用 PCA 来提取这些特征子空间的主成分,对于一个新的测试图像,简单的将它向各个特征子空间的主成分上进行投影,并计算投影后的残差。如果这个图像是某个特定物体的图像,那么它应该嵌入在这个特定物体的特征子空间上。忽略了微小的噪声后,它在这个特定的特征子空间上的投影残差应该为 0。基于这样的理论,测试图像在那个特征子空间的投影残差最小,它就是那个物体的图像。

然后流形学习理论添加到快速,用测地距离来代替 Guassian 核函数中的欧氏距离。测地距离是通过 3 部分距离来估算的,第 1 和第 3 部分是 2 个测试样本点到最近的 2 个流形学习样本点的欧氏距离,第 2 部分是 2 个流形学习样本点的测地距离。流形方法能够有效地提取每个类别的特征,极大地降低了特征子空间的维度。

1.1　基于 KPCA 的快速算法

一个物体的所有图像应该是一个无穷的集合 I_k,它包括了光照、角度、平移、距离等变化,其中,k 为物体的标签,$k = 1,2,3,\cdots,K$,K 为物体的总数目。根据前面叙述,有必要改变这些图像集合,使它们成为低维的、两两正交的特征子空间 FI_k。使用与 Gussian 核函数相对应的非线性特征映射 $\varphi: R^N \rightarrow F$ 将高维图像集合映射到特征空间中来实现这一过程,如图 1 所示。

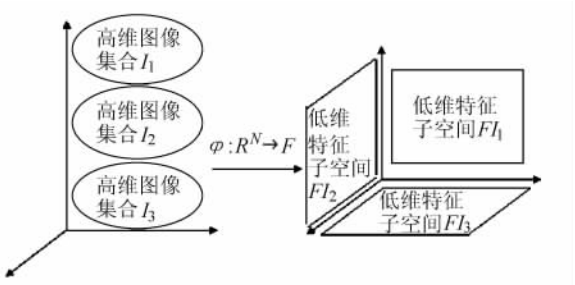

图 1　非线性特征映射将高维图像集合转化为低维、两两正交的特征子空间

Fig. 1　Nonlinear feature function transforms high dimensional image sets to low dimensional and orthogonal feature sub-spaces

Gaussian 函数的表达式如下:

$$\langle \varphi(x_1), \varphi(x_2) \rangle = k(x_1, x_2) = \exp\left(-\frac{\| x_1 - x_2 \|^2}{2\sigma^2}\right) \tag{1}$$

实际上 φ 的数学表达式是未知的,但是可以利用核函数直接计算 $\varphi(x_i)$ 之间的内积,这样就足可以进行 PCA 统计分析。通过非线性的特征映射,在特征空间中提取的线性特征,返回到原空间就是非线性特征。非线性特征映射随着 Gaussian 的参数 σ 的变化而变化,所以可以调节该参数来得到对最优的非线性特征映射。

可以看到,如果 x_1、x_2 是同一物体的图像,它们的欧氏距离应该比较近,那么 $\varphi(x_1)$、$\varphi(x_2)$ 的内积就会接近 1,由于 $\| \varphi(x_i) \| = 1$,所以它们很接近,可以认为处于一个维度上。相反的,如果 x_1、x_2 是不

同物体的图像,它们的距离相对较远,内积就会接近0,这就意味着不同物体的特征子空间趋向于正交。整个特征空间由 K 个特征子空间构成,如果各个特征子空间相互正交,当将测试图像向各个特征子空间投影时,不同特征子空间的投影残差的变化就会更剧烈,有利于进行分类。

另一方面,该非线性特征映射也对各个物体的图像集合进行了降维,特征子空间的维度不会超过训练样本的个数。x_i 的像 $\varphi(x_i)$ 实际上是一个广义函数,而特征空间 F 实际上是一个由 $\varphi(x_i)$ 张成的RKHS空间。一旦使用了核方法,原空间唯一保留下来的信息就是 $\varphi(x_i)$ 之间的内积,中心化的内积矩阵与距离矩阵是一一对应的,它决定了所有的空间结构。而 $\varphi(x_1)$、$\varphi(x_2)$ 之间的夹角为:

$$\frac{\langle \varphi(x_1), \varphi(x_2) \rangle}{\| \varphi(x_1) \| \cdot \| \varphi(x_2) \|} = k(x_1, x_2) = \exp\left(-\frac{\| x_1 - x_2 \|^2}{2\sigma^2}\right) \quad (2)$$

所以 Gaussian 核函数实际上将原空间中 x_1 附近的区域变成特征空间中的一个维度。实际上,可以认为 F 由训练样本的像 $\varphi(x_i)$ 张成,所以 F 的维度也不会超过训练样本的个数。这样就通过与 Gaussian 核函数相对应的特征映射建立了低维的、两两正交的特征子空间。

然后用 PCA 提取每个特征子空间的主成分。假设 $IS_k = \{x_{k,1}, x_{k,2}, \cdots, x_{k,l}\}$, $x_{k,l} \in R^N$ 为训练样本集,它是 I_k 的一个子集,l 表示训练样本的个数。使用非线性特征映射 φ 将 IS_k 映射到特征空间中 $FIS_k = \{\varphi(x_{k,1}), \varphi(x_{k,2}), \cdots, \varphi(x_{k,l})\}$,它是 FI_k 的一个子集。假设样本已经中心化,然后就在该特征图像子空间上做 PCA。如果训练样本覆盖了每个物体的图像的所有变化方向,PCA 就能够提取特征子空间 FI_k 的所有非线性特征。

每个特征子空间的协方差矩阵可以表示为:

$$\boldsymbol{C}_k = \frac{1}{l} \sum_{i=1}^{l} \varphi(x_{k,i}) \varphi(x_{k,i})^{\mathrm{T}} \quad (3)$$

需要寻找特征值 $\lambda_{k,j}$ 特征向量 $\boldsymbol{V}_{k,j}$ 使得

$$\lambda_{k,j} \boldsymbol{V}_{k,j} = \boldsymbol{C}_k \boldsymbol{V}_{k,j} \quad (4)$$

特征子空间由 $\varphi(x_{k,1}), \varphi(x_{k,2}), \cdots, \varphi(x_{k,l})$ 张成,所以 $\boldsymbol{V}_{k,j}$ 应该由它们的线性组合构成,将式(4)变形为:

$$\lambda(\varphi(x_{k,i}) \cdot \boldsymbol{V}_{k,j}) = (\varphi(x_{k,i}) \cdot \boldsymbol{C}_k \boldsymbol{V}_{k,j}) \quad (5)$$

其中,$k = 1, 2, \cdots, l$。

记 $m_{k,i,j} = \langle \varphi(x_{k,i}), \varphi(x_{k,j}) \rangle$,并将 $\boldsymbol{V}_{k,j} = \sum_i \alpha_{k,i,j} \varphi(x_{k,i})$ 代入得:

$$l\lambda \boldsymbol{A}_{k,j} = \boldsymbol{M}_k \boldsymbol{A}_{k,j} \quad (6)$$

其中,$\boldsymbol{M}_k$ 为由 $m_{k,i,j}$ 构成的矩阵,$\boldsymbol{A}_{k,j}$ 为由 $\alpha_{k,i,j}$ 构成的向量。这样就可以求出 $\boldsymbol{A}_{k,j}$,也就求出了 $\boldsymbol{V}_{k,j}$,提取了每个特征子空间的主成分后就可以计算测试图像到每个特征子空间的投影残差了。

对于测试图像 x,先使用非线性特征映射 φ 将其映射为 $\varphi(x)$,然后向每个特征子空间 FI_k 的主成分进行投射:

$$\beta_{k,j} = \langle \varphi(x), \boldsymbol{V}_{k,j} \rangle = \sum_i \alpha_{k,i,j} \langle \varphi(x), \varphi(x_{k,i}) \rangle \quad (7)$$

所以 $\varphi(x)$ 在每个特征子空间的投影为:

$$P_k \varphi(x) = \sum_j \beta_{k,j} \boldsymbol{V}_{k,j} = \sum_{ij} \alpha_{k,i,j} \beta_{k,j} \varphi(x_{k,i}) = \sum_i \gamma_{k,i} \varphi(x_{k,i}) \quad (8)$$

其中,$\gamma_{k,i} = \sum_j \alpha_{k,i,j} \beta_{k,j}$,然后就可以计算 $\varphi(x)$ 在每个特征子空间的投影残差,如图2所示。

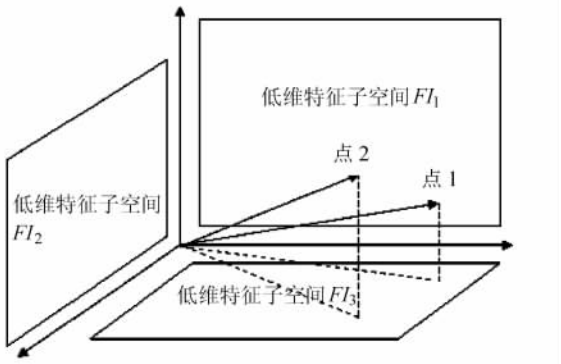

图2 投影残差的比较

Fig. 2 Comparison of projection residuals

$$PR_k = \| \varphi(x) - P_k \varphi(x) \|^2 = 1 - 2\sum_i \gamma_{k,i} \langle \varphi(x_{k,i}), \varphi(x) \rangle + \sum_{ij} \gamma_{k,i} \gamma_{k,j} \langle \varphi(x_{k,i}), \varphi(x_{k,j}) \rangle = 1 - \sum_i \gamma_{k,i} \langle \varphi(x_{k,i}), \varphi(x) \rangle \quad (9)$$

根据前面的分析,如果 x 是某个特定物体的图像,那么 $\varphi(x)$ 应该位于这个特定物体的特征子空间内。忽略了微小的噪声后,它在这个特定的特征子空间上的投影残差应该为0。所以认为 $\varphi(x)$ 在哪个特征子空间的投影残差最小,它就是哪个物体的图像,这就建立了投影残差分类器。

该算法的运算复杂度是 $O(\sum_k n_k)$,其中,n_k 为 IS_k 的训练样本个数。与 SVM 方法相比,这种算法有一个优点:它能一次区分多个类别。而 SVM 方法一次只能区分2个类别,对于一个 K 类问题,就需要

多次分类,而且多次分类的方法和次序都是待定的,这比较麻烦而且增加了不确定性。SVM 方法致力于在 2 类之间建立边界,而投影残差法则将各个类别放到一起,并把它们转化为低维的、两两正交的特征子空间,提取每个类别的特征,然后使用投影残差寻找最近的特征子空间,这一过程模仿了人类的思维模式。

该算法有一个特点: 当一个新的图像添加到训练样本集时,如果它没有被 KPCA 视为噪声,这个图像就能被正确识别。由于这个图像的映像与它的邻近图像的映像之间的内积接近于 1,它们在特征空间中实际上是在一个维度上。所以当这个新的图像被添加到训练样本时,特征子空间中就会添加体现这个图像邻域的主成分,这个图像以及跟它相类似的图像就会被正确的识别。如果训练样本能覆盖所有的图像变动方向,那么识别率就能够提高到 100%。

对于用来分类的投影残差的大小并没有固定的标准,像图 2 中样本点 1,它到特征子空间 FI_1 与 FI_3 的投影残差都很小,但样本点 2 到各个特征子空间的投影残差却比都较大。如果添加放缩因子,对于某一特定的分类问题也许能找到固定的标准,但这里只比较他们的相对大小。

1.2　基于 SDE 的改进算法

正如前面所述,虽然图像的维度非常高,但是图像集合的维度却比较低,最多只有光照、角度、平移、距离等几个维度,整个图像集合实际处在高维空间的一个低维流形上。虽然使用特征映射后虽然大大降低了维度,但效果并非很理想,而且识别率受训练样本覆盖率的影响较大。流形学习理论既然能识别集合的内在维度,如果将它添加到模型中,是否会提高识别率? 为了验证流形学习理论的实用性,将 SDE 方法引入到模型中,SDE 是目前比较经典的流形学习理论之一。假定每个流形学习样本只与周围的几个邻点相连接,SDE 在尽量保持邻点之间的距离与角度的同时,使所有流形学习样本两两之间的距离之和最大化。它保持了局部结构,而尽量展开整个流形。

在把图像集合转化为特征子空间之前,在每个集合中挑选它们各自的流行学习样本,使用 SDE 方法分别拉伸每个集合所处的流形,这样就发现了流形学习样本在流形上的坐标。在每个集合内部,两个样本之间的距离不在是欧氏距离而是测地距离,所以用流形上的测地距离来代替 Gaussian 核函数中的欧氏距离。在这里只是简单的估计测地距离,而没有使用具体的函数表达式。

在图 3 用 swiss-roll 为例来做说明测地距离的估计,图中黑色的点为流形学习样本,灰色的点为测试样本。左边的图是 swiss-roll 展开前的图形,右边是其流形展开后的图形。通过 3 部分来估计测地距离,第 1 和第 3 部分是测试样本点到最近的流形学习样本点的欧氏距离,第 2 部分是 2 个流形学习样本点的测地距离。如果流形学习样本点很致密,这个估计距离与真实的测地距离应该很接近。

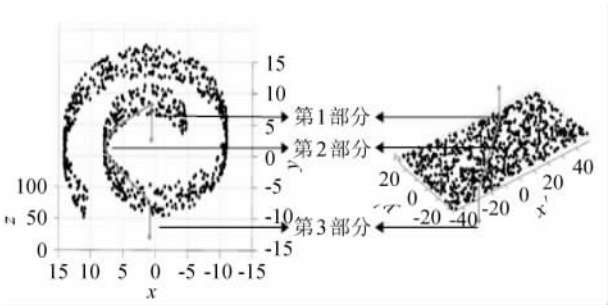

图 3　测地距离的估计

Fig. 3　Estimation of geodesic distance

所以新的核函数的表达式是:

$$\langle \varphi(x_1), \varphi(x_2) \rangle = k(x_1, x_2) = \exp\left(-\frac{\| x_1 - y_1 \|^2 + \mu \| m_1 + m_2 \|^2 + \| x_2 - y_2 \|^2}{2\sigma^2}\right) \tag{10}$$

其中,$x_1(x_2)$ 为测试样本,$y_1(y_2)$ 为离它们最近的流形学习样本。$m_1(m_2)$ 为 $y_1(y_2)$ 在流形上的坐标,m_1、m_2 之间的欧氏距离代表了 y_1、y_2 之间的测地距离。SDE 在尽量拉伸整个流行的同时,会使相连接点之间的欧式距离发生轻微的变化,所以引入参数 μ 来调节测地距离的权重:

$$\mu = \mu_0 \frac{D_1}{D_2} \tag{11}$$

μ_0 为常数,D_1 为训练样本之间的两两欧式距离和,D_2 为训练样本之间的两两测地距离和。

使用 SDE 方法极大地降低了特征子空间的维度,新坐标体现了光照、角度、平移、距离等变化,反映样本之间真实的测地距离。

引入了流形学习后,特征子空间的维度急剧减少。目前只优化了特征子空间内部的点积,却没对各个特征子空间做工作。如果能增强各个图像特征空间之间的正交性,新来的测试图像在各个特征子空间的投影残差会将更加剧烈变动,这将非常有利于进行分类。

2 实 验

为了验证算法,首先将算法在2个国际上通用的数据库上进行了测试。一个是coil-20数据库,一个是广泛流行的手写体USPS数据库。该算法可以应用于任何高维数据的分类,为了简单起见,只测试它在图像处理上的应用。

Coil-20包含20个物体的图像,每个物体放在一个转台上。相机每隔5°就拍一次照,所以每个物体都有72张图像。USPS数据库包含了9 298个手写体数字的图像,其中前7 291个图像是训练样本,后2 007个图像是测试样本。

2.1 快速算法的实验

对于Coil-20数据库,在每个物体的图像中均匀了取了1/8、1/4、1/3来做为训练样本,将所有图像作为测试样本运行快速算法,参数 $\sigma = 4.5 \times 10^7/256^2$,PCA显著率为98%,实验数据如表1所示。

表1 快速算法的Coil-20实验结果

Tab.1 Experiment results of fast algorithm for Coil-20

训练样本个数	特征子空间平均维度	错误个数	识别率/%	运行时间/s
9	7	39	97.29	0.052 5
18	16	10	99.31	0.099 3
24	22	0	100	0.131 6

对于USPS数据库,只在7 291个训练样本中对每个数字选取了具有代表性的100、200、200、300、400个图像作为训练样本,参数 $\sigma = 4.5 \times 10^7/256^2$,PCA显著率为98%,对2 007个测试样本的实验数据如表2所示。

表2 快速算法的USPS实验结果

Tab.2 Experiment results of fast algorithm for USPS

训练样本个数	特征子空间平均维度	错误个数	识别率/%	运行时间/s
100	75.5	128	93.87	0.025 2
200	124.3	94	95.32	0.042 8
300	159.4	85	95.77	0.051 8
400	186.6	83	95.87	0.109 6

针对同样的数据库,文献[4-6]以7 300个训练样本,对2 000个测试样本使用SVM的最好识别率为95.8%。文献[7]以3 000个训练样本,对2 000个测试样本使用LDA的最好识别率为96.3%。SVM与LDA都是针对2类别问题的分类器,这些文献虽然给出了对USPS数据库的实验结果,但并没有详细阐述多类问题的实际分类流程,这里只是引用它们的实验结果。

该算法的运算复杂度是 $O(\sum_k n_k)$,表1、2验证了运行时间与训练样本个数之间的线性关系,所以在识别率和运行时间此消彼长。实验中的训练样本都是手动选取的,如果能使用程序迭代选取,相信实验效果会更好。

当使用1/3的图像作为训练样本时,对coil-20的识别率达到了100%。对于USPS数据库,很多数字的图像已经完全被扭曲了,也有部分数字的图像太相像了,即使是人也很难区分。这验证该算法的特点:当一个新的图像添加到训练样本时,这个图像以及跟它相类似的图像就会被正确的识别。

计算5个USPS图像在各个特征子空间的投影残差,虽然都是0的手写体图像,但投影残差的变化却比较剧烈,它们在 FI_0 的投影残差分别为0.000 1、0.000 3、0.000 9、0.001 1、0.002 5,所以用固定的投影残差大小来进行识别比较困难,因此算法比较的是它们的相对大小。利用相对大小关系可以进一步给出每个图像属于各个类别的概率:

$$P_i = \frac{\sum_k PR_k/10 - PR_i}{\sum_k PR_k/10} \tag{12}$$

表3给出了实验结果。

表3 测试样本属于各类的概率

Tab.3 Probability that test sample belongs to each category

	0识别为0	4识别为4	6识别为6	3识别为5	4识别为9	8识别为0
P0	0.99	0.10	0.59	0.39	-0.72	0.71
P1	-4.77	-2.70	-2.12	-2.62	-1.09	-1.49
P2	0.83	0.37	0.79	0.40	-0.18	0.63
P3	0.82	0.23	0.57	0.86	0.26	0.70
P4	0.56	0.93	0.47	-0.08	0.77	0.09
P5	0.86	-0.07	0.70	0.87	0.11	0.62
P6	0.68	0.16	0.96	0.06	-0.94	0.60
P7	-0.79	0.14	-1.82	-0.55	0.58	-2.12
P8	0.72	0.31	0.59	0.74	0.43	0.66
P9	0.11	0.53	-0.73	-0.06	0.77	-0.39

当分类是正确的时候,只有一个类别的概率会比较大,而当分类错误时,会有多个不太大但却比较接近的概率。就像图2中的点1,它离 FI_1 和 FI_3 都比较近,在它们上的投影残差都比较小,相应的概率就会比较大而且比较接近。在各个特征子空间交界上的点比较难区分,但是自然界的样本并不一定充满各个特征子空间,所以这样的情况并非是必然出现的。当出现多个不太大的相近的概率时,错误分类的可能性就很大,可以用该方法进行预警。

特征子空间只是趋于正交,实际上所有的样本在特征空间中的模都等 1,它们处在一个单位球上,实验表明各个特征子空间的正交性很差。特征子空间的平均维度虽然大大低于图像的维度,但是由于非线性特征映射并没有专门针对于降维,特征子空间的降维效果并非很理想。在后期添加了流形学习理论后,降维效果非常好。

2.2　改进算法的实验

在实施例一中添加了流形学习理论。在 Coil-20 实验中,将每个物体的 72 个图像作为流形学习样本,用 SDE 分别对每个物体的图像进行流形展开。在 USPS 图像中,则使用每个数字的少于 1 200 个的图像作为流形学习样本。然后用改进的算法进行测试,参数 $\sigma = 6.5 \times 10^7 / 256^2$,$\mu_0 = 1.5$,PCA 显著率为 98%,实验数据如表 4、5 所示。

表 4　改进算法的 Coil-20 实验结果

Tab. 4　Experiment results of advanced algorithm for Coil-20

训练样本个数	特征子空间平均维度	错误个数	识别率/%
9	6.15	15	98.33
18	10.55	3	99.38
24	12.25	0	100

表 5　改进算法的 USPS 实验结果

Tab. 5　Experiment results of advanced algorithm for USPS

训练样本个数	特征子空间平均维度	错误个数	识别率/%
100	13.6	98	95.12
200	14.8	91	95.47
300	14.6	91	95.47
400	15.3	88	95.62

通过表 4、5 可以看到,对于数量较少的训练样本,流形学习理论能显著的提高识别率,这是因为 SDE 的流行学习样本提供了大量信息。对于数量较大的训练样本,识别率几乎没什么变化。这里只优化了每个特征子空间内部的关系,却没有增强各个特征子空间的正交性,相信在加强这方面工作后,实验效果将会有明显的提高。添加了流形学习理论后,特征子空间的维度大大降低了,每个特征子空间只需要 15 个维度,就能够达到 95.6% 的识别率。现在有很多流行的流形学习理论,但他们都没有验证这些理论的应用性。将流形学习理论添加到投影残差分类器中,取得较好的实验效果,这验证了流形学习理论在现实中的应用性。但是搜索离测试样本最近的流形学习样本花费时间较多,如果能建立直接的测地距离函数表达式,那么运行时间将会大大降低,甚至将比快速算法更快。

3　后期工作

虽然理论上特征子空间趋于两两正交,但实验数据表明它们的正交性比较差,在添加流行学习理论时也只是优化每个特征子空间内部的关系,没有直接优化特征子空间之间的正交性。有 2 条途径可以改善这个问题,一是直接优化核函数,二是提出多类流形学习方案,作者已经取得了更好的实验结果,将在后期的论文中详细阐述着两种方法,这里只给出理论框架。

改进的目标是使各个类别成为低维的、两两正交的特征子空间。尽量拉伸曲面流形,使得特征空间中每个类别的训练样本与这个类别的中心的距离最大化,以此降低特征子空间的维度,即:

$$\left\langle \varphi(x_{k,i}) - \frac{1}{n_k}\sum_j \varphi(x_{k,j}), \varphi(x_{k,i}) - \frac{1}{n_k}\sum_j \varphi(x_{k,j}) \right\rangle \to \infty \tag{13}$$

以特征空间中每个类别的训练样本与这个类别的中心来确定方向,使不同特征子空间的维度方向相互正交,以此来提高它们的正交性,即:

$$\left\langle \varphi(x_{k,i}) - \frac{1}{n_k}\sum_j \varphi(x_{k,j}), \varphi(x_{h,g}) - \frac{1}{n_h}\sum_j \varphi(x_{h,j}) \right\rangle \to 0 \tag{14}$$

中心化的内积矩阵决定了距离矩阵,也决定了样本点的所有空间结构,所以保持局部内积能保持局部结构,如果 x_e、x_h(e 可以等于 h) 都是 x_i 的邻近点,则

$$\left\langle \varphi(x_e) - \varphi(x_i), \varphi(x_h) - \varphi(x_i) \right\rangle \approx \left\langle x_e - x_i, x_h - x_i \right\rangle \tag{15}$$

以降低维度、提高正交性为目标,保持局部内积为约束条件,来优化核函数。采用的核函数表达式为 $k(x,y) = q(x)\,q(y)\,k_0(x,y)$,其中,$k_0(x,y)$ 为基础核函数,例如 Gaussian 核函数,

$$q(x) = \alpha_0 + \sum_i \alpha_i k_1(x, a_i) ,$$

$$k_1(x, a_i) = \exp(-\gamma \| x - a_i \|^2) ,$$

γ 与 a_i 为常数,α_i 为待定系数。

以降低维度为目标,提高正交性、保持局部内积为约束条件,来建立一种完全基于内积的多类别流形学习方法。然后搜索测试样本的多个邻近的训练样本,寻找这几个邻近的训练样本在特征空间中的线性组合,以保持局部距离为目标计算测试样本流形展开后的坐标,直接在流形学习建立的特征空间中用 PCA 进行投影残差分类。

4 结 论

提出了基于KPCA的快速算法和基于SDE的改进算法2种方法。与传统的分类方法不同,这2种算法能够一次区分多个类别。它们分别使用非线性特征映射以及添加了流形学习理论的非线性特征映射将各个类别转化为低维的两两正交的特征子空间,然后使用投影残差进行分类,取得很好的实验效果。

参考文献:

[1] Duda R O,Hart P E,Stork D G. Pattern classification [M]. 2nd ed. New York: Wiley,2000.

[2] Cover T M,Hart P E. Nearest neighbor pattern classification [J]. IEEE Transactions on Information Theory,1967,13(1):21 - 27.

[3] Freund Y,Schapire R E. A decision-theoretic generalization of on-line learning and an application to boosting [C]. Computational Learning Theory: Second European Conference, Berlin: Springer,1995,904: 23 - 37.

[4] Boser B E,Guyon I M,Vapnik V N. A training algorithm for optimal margin classifiers [C]//Proceedings of the Fifth Annual Workshop on Computational Learning Theory. New York: ACM,1992.

[5] Scholkopf B,Burges C ,Vapnik V. Extracting support data for a given task [C]. First International Conference on Knowledge Discovery & Data Mining,Menlo Park,California: AAAI,1995: 252 - 257.

[6] Burges C J C. Simplified support vector decision rules [C]. 13th International Conference on Machine Learning, Waltham,Massachusetts: Morgan Kaufman, 1996: 71 - 77.

[7] Mika S,Ratsch G,Weston J,et al. Fisher discriminant analysis with kernels [C]//Neural Networks for Signal Processing. New York: IEEE,1999,IX: 41 - 48.

[8] Vapnik V. The nature of statistical learning theory [M]. New York: Springer Verlag,1995.

[9] Cox T F,Cox M A A. Multidimensional scaling [M]. New York: CRC,2001.

[10] Roweis S T,Saul L K. Nonlinear dimensionality reduction by local linear embedding [J]. Science,2000,390: 2323 - 2326.

[11] Tenenbaum J B,de Silva V,Langford J C. A global geometric framework for nonlinear dimensionality reduction [J]. Science,2000,390: 2319 - 2323.

[12] Weinberger K Q,Sha Fei,Saul L K. Learning a kernel matrix for nonlinear dimensionality reduction [C]//Proceedings of the Twenty-first International Conference on Machine Learning. New York: ACM,2004: 106.

[13] Weinberger K Q,Saul L K. Unsupervised learning of image manifolds by semidefinite programming [C]//Computer Vision and Pattern Recognition. New York: IEEE,2004,2: 988 - 995.

[14] Belkin M,Niyogi P. Laplacian eigenmaps for dimensionality reduction and data representation [J]. Neural Computation, 2003,15: 1373 - 1396.

[15] Schölkopf B ,Smola A ,Müller K-R. Kenel principle component analysis [C]//Artificial Neural Networks. Berlin: Springer,1997,1327: 583 - 588.

[16] Bengio Y,Delalleau O,Paiement J-F,et al. Learning eigenfunctions links spectral embedding and kernel PCA [J]. Neural Computation,2004,16(10): 2197 - 2219.

[17] Tatjun C,Suter D. Out-of-sample extrapolation of learned manifolds [J]. Pattern Analysis and Machine Intelligence, 2008,30(9): 1547 - 1556.

[18] Xiong Huilin,Swamy M N S,Ahmad M O. Optimizing the kernel in the empirical feature space [J]. Neural Network, 2005,16(2): 460 - 474.

[19] Chen Bo,Liu Hongwei,Bao Zheng. Optimizing the data-dependent kernel under a unified kernel optimization frame work [J]. Pattern Recognition,2008,41(6): 2107 - 2119.

[20] Shaoa Jidong,Ronga Gang,Leeb Jong Min. Learning a data-dependent kernel function for KPCA-based nonlinear process monitoring [J]. Chemical Engineering Research and Design, 2009,87(11): 1471 - 1480.

(编辑 杨 蓓)